

Amendments to Claims

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims

1. (currently amended) A method for accelerating access to data on a network comprising:
 - providing a plurality of computers on the network, each with cache software;
 - receiving in one of the computers, from an application, a write instruction that specifically addresses data in a cached I/O device connected to the network;
 - maintaining exclusive write access control over the specifically addressed data in the cached I/O device;
 - writing data into cache in the one of the computers responsive to the write instruction;
 - communicating over the network with the cache software at all computers that permit caching with respect to the cached I/O device to invalidate data in remote caches on the network that cache the cached I/O device;
 - sending a write I/O completion signal to the application after completing the invalidation of the specifically addressed data in the remote caches on the network.
2. (original) The method of claim 1 wherein the specifically addressed data is a single data block.
3. (original) The method of claim 1 further comprising releasing the exclusive write access control after completing the invalidation of the specifically addressed data in the remote caches on the network.

4. (currently amended) A method for coherently caching a shared I/O device available on a network comprising:

providing a plurality of computers on the network each with cache software; and
creating a data structure that provides a list of the computers on the network that permit caching with respect to the shared I/O device; and

after data is written to the shared I/O device, communicating with each computer in the list of computers to invalidate data in remote caches corresponding to the shared I/O device.

5. (canceled)

6. (canceled)

7. (previously presented) A method for accelerating access to data on a network comprising:

providing a plurality of computers on the network, each with cache software;
receiving, in one of the computers, a write instruction having data to be written into addresses on one of said I/O devices;

selecting one of a plurality of cache data bucket sizes available in the one of the computers;

writing the data into a data bucket of the selected data bucket size responsive to the write instruction; and

communicating over the network with remote caches to invalidate cache data corresponding to any of the addresses.

8. (original) The method of claim 7 wherein a byte count of the data written responsive to the write instruction determines the cache data bucket size selected.

9. (original) A method for coherently caching an I/O device available for shared access on a network comprising:

providing a plurality of computers on the network, each with cache software; and
privately communicating on a computer communication channel between cache software on a first of the computers caching the I/O device and cache software on a second of the computers caching the I/O device.

10. (original) The method of claim 9 further comprising intercepting, in the cache software of the first of the computers, a write instruction to the I/O device.

11. (original) The method of claim 10 wherein privately communicating comprises sending a message through the computer communication channel from the cache software of the first of the computers to the cache software of the second of the computers to invalidate data responsive to the write instruction.

12. (original) The method of claim 11 further comprising privately communicating on computer communication channels with all of the computers caching the I/O device to invalidate data responsive to the write instruction.

13. (original) A method for coherently caching I/O devices available for shared access on a network comprising:

providing a plurality of computers on the network, each with cache software, the cache software for storing data into any of a plurality of caches each being of a different data bucket size;

receiving a write instruction having data to be written into addresses in one of the I/O devices;

writing data into one of the caches responsive to the write instruction; and

communicating over the network to invalidate cache data corresponding to addresses that overlap with addresses for the data in the write instruction.

14. (original) A method for coherently caching I/O devices available for shared access on a network comprising:

providing a network with a plurality of nodes, each with a cache driver;
creating a cache for one of the I/O devices at one of the nodes not currently caching said one of the I/O devices; and

communicating with all the cache drivers at all the nodes on the network that permit caching with respect to said one of the I/O devices to inform them that said one of the I/O devices may be cached on said one of the nodes.

15. (currently amended) A method for coherently caching I/O devices available for shared access on a network comprising:

providing a network with a plurality of nodes, each with a cache driver; and

registering each cached I/O device with all the cache drivers on all nodes that ~~each~~ permit caching said cached I/O device.

16. (original) The method of claim 15 further comprising forming communication channels with each of the nodes in said plurality of nodes.

17. (original) A method for coherently caching I/O devices available for shared access on a network, comprising:

providing a network with a plurality of nodes, each with a cache driver; and

creating, at each node that may cache one of the I/O devices, a data structure for the one of the I/O devices that includes a list of all nodes on said network that permit caching with respect to the one of the I/O devices.

18. (new) The method of claim 17 further comprising:

receiving an instruction to write data to a shared cached disk I/O device in one of the nodes;

writing said data into the cache in the one of the nodes that received the write instruction;

causing the writing of cached data to be written to the shared cached I/O device;

receiving back the write completion signal from the shared cached I/O device, after the data has been written to it; and

communicating over the network to invalidate the data block(s), corresponding to any address just written to in the shared cached I/O device, in the caches on the remote nodes listed in the list in the cache driver for the shared cached I/O device.

19. (new) The method of claim 18 wherein the data written to the shared cached I/O device and invalidated on the remote nodes in the list is a single data block.

20. (new) The method of claim 18 further comprising:

disabling cache operations upon finding that a new node joined the network; and

enabling caching operations at each node after each node has connections in place with the cache driver of every other node on the network.

21. (new) The method of claim 18 further comprising listening on the network for a request from a new node to join the network.

22. (new) The method of claim 18 wherein said communicating over the network occurs after said receiving back the write completion signal.

23. (new) A method for accelerating access to data on a network comprising:
providing a plurality of computers on the network, each with cache software;
causing the writing of cached data block(s) to be written to a shared cached I/O device;

receiving back a write completion signal from the shared cached I/O device, after the data has been written to it; and

communicating over the network to invalidate the data block(s), corresponding to the data just written to the shared cached I/O device, in the remote caches on the network listed in a list for the shared cached I/O device provided by the cache software, said list corresponding to the caches on the remote nodes that permit caching with respect to the shared cached I/O device.

24. (new) The method of claim 23 wherein the data written to shared cached I/O device and invalidated at the remote nodes in the list is a single data block.

25. (new) The method of claim 23 further comprising:
disabling cache operations upon finding that a new computer joined the network; and
enabling caching operations at each computer after each computer has connections in place with the cache software of every other computer on the network.

26. (new) The method of claim 23 further comprising listening on the network for a request from a new computer to join the network.

27. (new) The method of claim 23 wherein said communicating over the network occurs after said receiving back the write completion signal.

28. (new) The method of claim 4 further comprising:

disabling caching operations upon finding that a new computer has joined the network; and

enabling caching operations at each computer after each computer has connections in place with the cache software of every other computer in the network.

29. (new) The method of claim 28 further comprising listening on the network for a request from a new computer to join the network.

30. (new) The method of claim 7 further comprising receiving back a write completion signal from the one of said I/O devices and wherein communicating over the network occurs after said receiving back.

INTERVIEW SUMMARY

On March 15, 2005, Applicant's counsel met with Examiner Namazi to review a number of the facts and positions set forth below. The distinction between an experimental use and a public use was discussed along with the factual statement that follows in the Remarks. The issue of whether the SuperCache Guide constitutes a printed publication in view of the Federal Circuit decision in the *Northern Telecom* case was discussed. The obstacles to implementing multiple bucket size caches on a network of computers was reviewed. We further discussed the distinction between the targeting of invalidate messages to particular nodes or computers and broadcasting invalidates to all nodes or computers on a network. The Examiner agreed to consider the response in detail when filed and expressed no disagreement with the positions offered by Applicant's counsel. The Examiner is encouraged to contact Applicant's counsel with any questions that may remain upon full consideration of this response.

Since the interview, the response has been revised, edited and supplemented with the Declaration of Eric Dickman, the addition of claims 22, 27 and 30, a discussion of how I/O Express sends invalidates before sending data to disk as it relates to claim 4 and the new claims 22, 27 and 30.